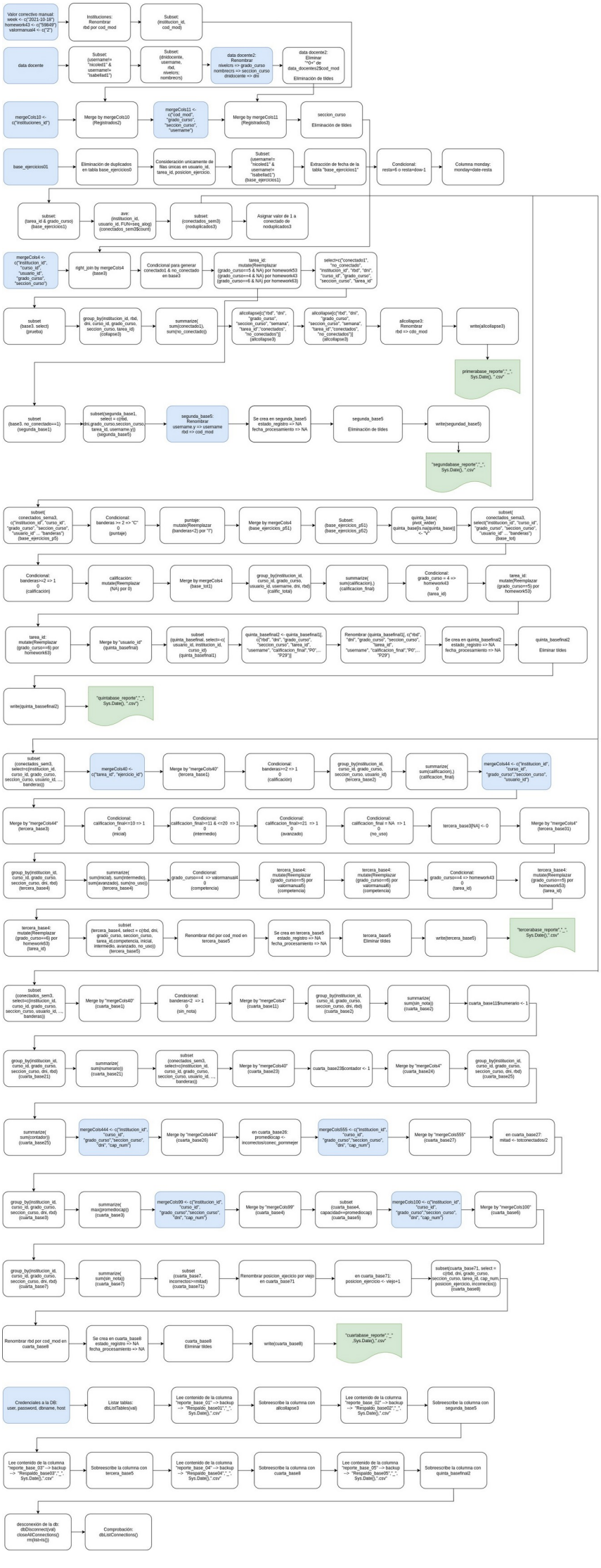
Valor correctivo manual: week <- c("2021-10-18") homework43 <- c("59849") valormanual4 <- c("2")
Instituciones: Renombrar rbd por cod_mod
Subset: (institucion_id, cod_mod)
data docente
Subset: (username!= "nicoled1" & username!= "isabella01")
Subset: (dnidocente, username, rbd, nivelcrs, nombrecrs)
data docente2: Renombrar nivelcrs => grado_curso nombrecrs => seccion_curso dnidocente => dni
data docente2: Eliminar "0+" de data_docentes2$cod_mod Eliminación de tildes
mergeCols10 <- c("instituciones_id")
Merge by mergeCols10 (Registrados2)
mergeCols11 <- c("cod_mod", "grado_curso", "seccion_curso", "username")
Merge by mergeCols11 (Registrados3)
seccion_curso Eliminación de tildes
base_ejercicios01
Eliminación de duplicados en tabla base_ejercicios0
Consideración unicamente de filas únicas en usuario_id, tarea_id, posicion_ejercicio.
Subset: (username!= "nicoled1" & username!= "isabella01") (base_ejercicios1)
Extracción de fecha de la tabla "base_ejercicios1"
Condicional: resta=6 o resta=dow-1
Columna monday: monday=date-resta
subset: (tarea_id & grado_curso) (base_ejercicios1)
ave: (institucion_id, usuario_id, FUN=(seq_along) (conectados_sem3$count)
subset: (conectados_sem3) (noduplicados3)
Asignar valor de 1 a conectado de noduplicados3
mergeCols4 <- c("institucion_id", "curso_id", "usuario_id", "grado_curso", "seccion_curso")
right_join by mergeCols4 (base3)
Condicional para generar conectado1 & no_conectado en base3
tarea_id: mutate(Reemplazar (grado_curso==5 & NA) por homework53 (grado_curso==4 & NA) por homework43 (grado_curso==6 & NA) por homework63)
select=c("conectado1", "no_conectado", "institucion_id", "rbd", "dni", "curso_id", "grado_curso", "seccion_curso", "tarea_id"
subset (base3, select) (prueba)
group_by(institucion_id, rbd, dni, curso_id, grado_curso, seccion_curso, tarea_id) (colapse3)
summarize(sum(conectado1), sum(no_conectado))
alicollapse(c("rbd", "dni", "grado_curso", "seccion_curso", "semana", "tarea_id","conectado", "no_conectado")) (alicollapse3)
alicollapse(c("rbd", "dni", "grado_curso", "seccion_curso", "semana", "tarea_id","conectado", "no_conectado")) (alicollapse3)
alicollapse3: Renombrar rbd => cdo_mod
write(alicollapse3)
primerabase_reporte"+"_"+ Sys.Date(), ".csv" ~
subset (base3, no_conectado==1) (segunda_base1)
subset(segunda_base1, select = c(rbd, dni,grado_curso,seccion_curso, tarea_id, username,y)) (segunda_base5)
segunda_base5: Renombrar username y => username rbd => cod_mod
Se crea en segunda_base5 estado_registro => NA fecha_procesamiento => NA
segunda_base5 Eliminación de tildes
write(segundad_base5)
"segundabase_reporte"+"_"+ Sys.Date(), ".csv"
subset(conectados_sema3, c("institucion_id", "curso_id", "grado_curso", "seccion_curso", "usuario_id" ... "banderas") (base_ejercicios_p5)
Condicional: banderas >= 2 => "C" 0 (puntaje)
puntaje: mutate(Reemplazar (banderas<2) por "I")
Merge by mergeCols4 (base_ejercicios_p51)
Subset: (base_ejercicios_p51) (base_ejercicios_p52)
quinta_base(pivot_wider) quinta_base[is.na(quinta_base)] <- "V"
subset(conectados_sema3, select("institucion_id", "curso_id", "grado_curso", "seccion_curso", "usuario_id" ... "banderas") (base_tot)
Condicional: banderas>=2 => 1 0 (calificación)
calificacion: mutate(Reemplazar (NA) por 0)
Merge by mergeCols4 (base_tot1)
group_by(institucion_id, curso_id, grado_curso, usuario_id, username, dni, rbd) (calific_total)
summarize(sum(calificacion,) (calificacion_final)
Condicional: grado_curso = 4 => homework43 0 (tarea_id)
tarea_id: mutate(Reemplazar (grado_curso==5) por homework53)
tarea_id: mutate(Reemplazar (grado_curso==6) por homework63)
Merge by "usuario_id" (quinta_basefinal)
subset (quinta_basefinal, select=-c(usuario_id, institucion_id, curso_id) (quinta_basefinal1)
quinta_basefinal2 <- quinta_basefinal1[, c("rbd", "dni", "grado_curso", "seccion_curso", "tarea_id", "username", "calificacion_final", "P0",... "P29")]
Renombrar (quinta_basefinal1, c("rbd", "dni", "grado_curso", "seccion_curso", "tarea_id", "username", "calificacion_final","P0",... "P29")
Se crea en quinta_basefinal2 estado_registro => NA fecha_procesamiento => NA
quinta_basefinal2 Eliminar tildes
write(quinta_basefinal2)
"quintabase_reporte"+"_"+ Sys.Date(), ".csv"
subset (conectados_sem3, select=c(institucion_id, curso_id, grado_curso, seccion_curso, usuario_id, ... banderas))
mergeCols40 <- c("tarea_id", "ejercicio_id")
Merge by "mergeCols40" (tercera_base1)
Condicional: banderas>=2 => 1 0 (calificación)
group_by(institucion_id, curso_id, grado_curso, seccion_curso, usuario_id) (tercera_base2)
summarize(sum(calificacion,) (calificacion_final)
mergeCols44 <- c("institucion_id", "curso_id", "grado_curso","seccion_curso", "usuario_id")
Merge by "mergeCols44" (tercera_base3)
Condicional: calificacion_final<=10 => 1 0 (inicial)
Condicional: calificacion_final>=11 & <=20 => 1 0 (intermedio)
Condicional: calificacion_final>=21 => 1 0 (avanzado)
Condicional: calificacion_final = NA => 1 0 (no_uso)
tercera_base3[NA] <- 0
Merge by "mergeCols4" (tercera_base31)
group_by(institucion_id, curso_id, grado_curso, seccion_curso, dni, rbd) (tercera_base4)
summarize(sum(inicial), sum(intermedio), sum(avanzado), sum(no_uso)) (tercera_base4)
Condicional: grado_curso==4 => valormanual4 0 (competencia)
tercera_base4: mutate(Reemplazar (grado_curso==5) por valormanual5) (competencia)
tercera_base4: mutate(Reemplazar (grado_curso==6) por valormanual6) (competencia)
Condicional: grado_curso==4 => homework43 0 (tarea_id)
tercera_base4: mutate(Reemplazar (grado_curso==5) por homework53) (tarea_id)
tercera_base4: mutate(Reemplazar (grado_curso==6) por homework63) (tarea_id)
subset (tercera_base4, select = c(rbd, dni, grado_curso, seccion_curso, tarea_id,competencia, inicial, intermedio, avanzado, no_uso)) (tercera_base5)
Renombrar rbd por cod_mod en tercera_base5
Se crea en tercera_base5 estado_registro => NA fecha_procesamiento => NA
tercera_base5 Eliminar tildes
write(tercera_base5)
"tercerabase_reporte"+"_"+ Sys.Date(), ".csv"
subset (conectados_sem3, select=c(institucion_id, curso_id, grado_curso, seccion_curso, usuario_id, ... banderas))
Merge by "mergeCols40" (cuarta_base1)
Condicional: banderas<2 => 1 0 (sin_nota)
Merge by "mergeCols4" (cuarta_base11)
group_by(institucion_id, curso_id, grado_curso, seccion_curso, dni, rbd) (cuarta_base2)
summarize(sum(sin_nota)) (cuarta_base2)
cuarta_base11$numerario <- 1
group_by(institucion_id, curso_id, grado_curso, seccion_curso, dni, rbd) (cuarta_base21)
summarize(sum(numerario)) (cuarta_base21)
subset (conectados_sem3, select=c(institucion_id, curso_id, grado_curso, seccion_curso, usuario_id, ... banderas))
Merge by "mergeCols40" (cuarta_base23)
cuarta_base23$contador <- 1
Merge by "mergeCols4" (cuarta_base24)
group_by(institucion_id, curso_id, grado_curso, seccion_curso, dni, rbd) (cuarta_base25)
summarize(sum(contador)) (cuarta_base25)
mergeCols444 <- c("institucion_id", "curso_id", "grado_curso","seccion_curso", "dni", "cap_num")
Merge by "mergeCols444" (cuarta_base26)
en cuarta_base26: promediocap <- incorrectos/correc_porrmejer
mergeCols555 <- c("institucion_id", "curso_id", "grado_curso","seccion_curso", "dni", "cap_num")
Merge by "mergeCols555" (cuarta_base27)
en cuarta_base27: mitad <- totconectados/2
group_by(institucion_id, curso_id, grado_curso, seccion_curso, dni, rbd) (cuarta_base3)
summarize(max(promediocap)) (cuarta_base3)
mergeCols99 <- c("institucion_id", "curso_id", "grado_curso","seccion_curso", "dni", "cap_num")
Merge by "mergeCols99" (cuarta_base4)
subset (cuarta_base4, capacidad==promediocap) (cuarta_base5)
mergeCols100 <- c("institucion_id", "curso_id", "grado_curso","seccion_curso", "dni", "cap_num")
Merge by "mergeCols100" (cuarta_base6)
group_by(institucion_id, curso_id, grado_curso, seccion_curso, dni, rbd) (cuarta_base7)
summarize(sum(sin_nota)) (cuarta_base7)
subset (cuarta_base7, incorrectos>=mitad) (cuarta_base71)
Renombrar posicion_ejercicio por viejo en cuarta_base71
en cuarta_base71: posicion_ejercicio <- viejo+1
subset(cuarta_base71, select = c(rbd, dni, grado_curso, seccion_curso, tarea_id, cap_num, posicion_ejercicio, incorrectos)) (cuarta_base8)
Renombrar rbd por cod_mod en cuarta_base8
Se crea en cuarta_base8 estado_registro => NA fecha_procesamiento => NA
cuarta_base8 Eliminar tildes
write(cuarta_base8)
"cuartabase_reporte"+"_"+ Sys.Date(), ".csv"
Credenciales a la DB: user, password, dbname, host
Listar tablas: dbListTables(val)
Lee contenido de la columna "reporte_base_01" --> backup --> "Respaldo_base01"+"_"+ Sys.Date(),".csv"
Sobreescribe la columna con alicollapse3
Lee contenido de la columna "reporte_base_02" --> backup --> "Respaldo_base02"+"_"+ Sys.Date(),".csv"
Sobreescribe la columna con segunda_base5
Lee contenido de la columna "reporte_base_03" --> backup --> "Respaldo_base03"+"_"+ Sys.Date(),".csv"
Sobreescribe la columna con tercera_base5
Lee contenido de la columna "reporte_base_04" --> backup --> "Respaldo_base04"+"_"+ Sys.Date(),".csv"
Sobreescribe la columna con cuarta_base8
Lee contenido de la columna "reporte_base_05" --> backup --> "Respaldo_base05"+"_"+ Sys.Date(),".csv"
Sobreescribe la columna con quinta_basefinal2
desconexión de la db: dbDisconnect(val) closeAllConnections() rm(list=ls())
Comprobación: dbListConnections()